C++ ASSIGNMENT

1.Ques :Print all the elements of an array in reverse order.

Ans:

```
#include<iostream>
using namespace std;
void rever(int arr[],int i,int j){
        if(i>j) return;
   swap(arr[i],arr[j]);
        rever(arr,i+1,j-1);
}
int main(){
        int v[]={1,4,6,8,9};
        rever(v,0,4);
        for(int i=0;i<5;i++){
                cout<<v[i]<<" ";
        }
}
```

2.Ques :Print index of a given element in an array. If not present, print -1.

Ans:

```
#include<iostream>
#include<vector>
using namespace std;
void find(vector<int> arr,int lo,int hi,int target){
        if(lo>hi){
                cout<<-1;
                return;
        }
        int mid=lo+(hi-lo)/2;
```

```
        if(arr[mid]>target) find(arr,lo,mid-1,target);
        else if(arr[mid]==target){
                cout<<mid;
                return;
        }
        else find(arr,mid+1,hi,target);

}
int main(){
        int n;
        cin>>n;
        vector<int> v(n);
        for(int i=0;i<n;i++){
                cin>>v[i];
        }
        int x;
        cout<<"Enter the target element.";
        cin>>x;
        find(v,0,n-1,x);
}
```

3.Ques :A function countAndSay is defined as:

countAndSay(1) = “1”
countAndSay(n) is the way you would "say" the digit string from countAndSay(n-1), which is then converted into a different digit string.
So, if sample input is n = 4,
countAndSay(1) = 1
countAndSay(2) = “one 1” => 11
countAndSay(3) = “two 1” => 21
countAndSay(4) = “one 2 one 1” => 1211

Ans:
```
#include<iostream>
using namespace std;
string countAndSay(int n) {
    if(n==1) return"1";
    string ans="";
    string smallans=countAndSay(n-1);
    for(int i=0;i<smallans.size();){
      int count=1;
      int j=i+1;
      while(j<smallans.size()&&  smallans[i]==smallans[j]){
        j++;
        count++;
      }
      ans+=to_string(count)+smallans[i];
      i=j;
    }
    return ans;
}
int main(){
     cout<<countAndSay(4);
}
```

4.Ques :Given an array of integers, print a sum triangle using recursion from it such that the first level has
all array elements. After that, at each level the number of elements is one less than the previous
level and elements at the level will be the sum of consecutive two elements in the previous level.
So, if sample input is [5, 4, 3, 2, 1], sample output will be:
[5, 4, 3, 2, 1]
[9, 7, 5, 3]

[16, 12, 8]
[28, 20]
[48]

Ans:

```
#include<iostream>
#include<vector>
using namespace std;
void sumtriangle(vector<int> arr,int n){
	if(n==0) return;
	vector<int> temp(n-1);
	for(int i=0;i<n;i++){
		cout<<arr[i]<<" ";
		if(i!=0){
			temp[i-1]=arr[i]+arr[i-1];
		}
	}
	cout<<endl;
	sumtriangle(temp,n-1);

}
int main(){
	vector<int> v={5,4,3,2,1};
	int n=5;
	sumtriangle(v,n);
}
```